npj | metamaterials

Article

https://doi.org/10.1038/s44455-025-00015-4

Multimodal oscillator networks learn to solve a classification problem

Check for updates

Daan de Bos & Marc Serra-Garcia ✉

We numerically demonstrate a network of coupled oscillators that can learn to solve a classification task from a set of examples—performing both training and inference through the nonlinear evolution of the system. We accomplish this by combining three key elements to achieve learning: A long-term memory that stores learned responses, analogous to the synapses in biological brains; a short-term memory that stores the neural activations, similar to the firing patterns of neurons; and an evolution law that updates the synapses in response to novel examples, inspired by synaptic plasticity. Achieving all three elements in wave-based information processors such as metamaterials is a significant challenge. Here, we solve it by leveraging the material multistability to implement long-term memory, and harnessing symmetries and thermal noise to realize the learning rule. Our analysis reveals that the learning mechanism, although inspired by synaptic plasticity, also shares parallelisms with bacterial evolution strategies, where mutation rates increase in the presence of noxious stimuli.

Wave-based computing offers the tantalizing prospect of massive parallelism[1], speed-of-light operation[2] and low energy consumption[3]. Applications span a broad range of problems, including evaluating arithmetic operations[4], computing derivatives[5], solving integral equations[6], performing voice recognition[7,8], image classification[9,10], and generating pictures through reverse diffusion processes[11,12]. Traditionally, metamaterials that compute with waves are designed for fixed tasks via top-down algorithms such as adjoint optimization[13,14] or combinatorial methods[15]. This top-down approach contrasts with the continuous learning process that is characteristic of living brains, in which connections between decentralized neurons are autonomously formed using local information[16–18]. This decentralized learning process confers a resilience and adaptability that artificial systems lack, and reduces training costs.

Physical learning is an emerging field that aims at bringing the adaptability of biological organisms to artificial materials, by creating synthetic systems that learn from examples. A key strategy in physical learning consists in updating the model parameters according to a learning rule. These local learning rules are inspired by the synaptic plasticity mechanisms in neurons, with Hebbian learning[16,17]—the notion that neurons form connections based on correlations between firing patterns—being one of the foremost examples. Yet learning strategies also include many recent additions, such as equilibrium propagation[19], the forward-forward algorithm[20] and Hamiltonian echo backpropagation[21]. While much of the work has focused on developing and analyzing learning rules in abstract or algorithmic terms, an equally important frontier lies in the physical realization of systems that embody these rules. In this context, self-learning systems have been realized by externally updating the system parameters according to a learning rule, either manually[22] or using electronic circuits[23].

An open problem in physical learning is figuring out how to build physical systems where the learning occurs through the natural evolution of the material, instead of relying on manual updates or external electronic circuitry. Such material could potentially extend the parallelism and speed of wave computing to the learning process, enabling the use of physical phenomena to train machine learning models. However, realizing self-learning in metamaterials is a significant challenge. This is because a self-learning material is tasked with solving two simultaneous problems[24]: On one hand, the material performs a direct problem on the input signals. This direct problem can be, for example, solving a classification task. On the other hand, the material is also performing an inverse problem, autonomously adapting its parameters to improve its fitness in response to new examples.

Results

In this work, we introduce a metamaterial model that learns from examples. The metamaterial, consisting of a network of coupled multimodal resonators, simultaneously solves both learning and inference problems using wave physics. When solving a classification task, the proposed metamaterial can be seen as a square lattice of masses and springs. The input is applied as a harmonic force in one or several sites. In a flower classification problem, this input may consist of the geometric features of the flower, such as petal width and petal length—with higher amplitudes corresponding to larger feature values. The output of the computation is the vibration amplitude of a designated output site (Fig. 1a). Although spring networks (and mathematically analogous

AMOLF Science Park 104, Amsterdam, The Netherlands. ✉e-mail: m.serragarcia@amolf.nl

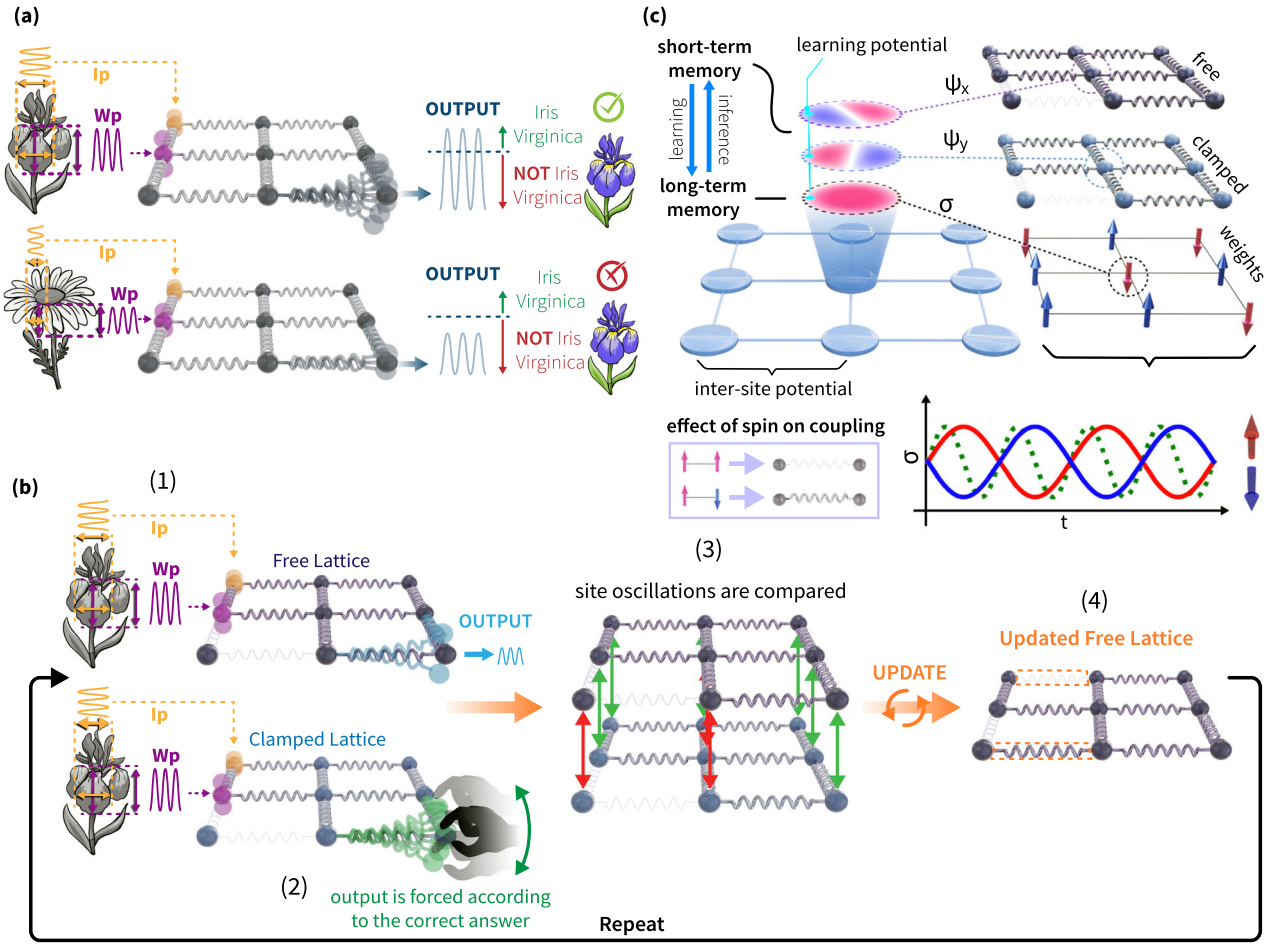

Fig. 1 | Self-learning metamaterial. a During inference, the metamaterial acts as a mass-spring network. Features are encoded in the excitation amplitude at designated input sites (yellow and purple spheres). If the output amplitude exceeds a threshold, an object is recognized. **b** For the material to learn according to a contrastive learning rule, an identical copy to the lattice in (**a**) is constructed. The original lattice is operated normally, letting the output site vibrate freely, and is referred to as the free lattice. In the copy, the output is clamped to vibrate at the correct amplitude. At every learning iteration, the coupling springs are updated depending on the difference between the vibrating amplitude of the clamped and free lattices. **c** In the proposed metamaterial, the masses of the free and clamped lattices are mapped to normal modes of each metamaterial site. We refer to these as ψ_x and ψ_y respectively (collectively referred as computational degrees of freedom). The coupling springs are encoded in an additional mode, that is represented by σ. This mode is parametrically driven at twice its natural frequency (dotted green line) and consequently has two stable phases of oscillation (represented as spin states with blue and red arrows). As a consequence of a nonlinear interaction potential, the effective springs connecting the computational modes depend on the weight state---taking a weak value when neighboring strings are aligned, and a strong value when neighboring weights are oscillating in opposite phases. The learning potential (blue line connecting the modes) causes the weight state to change when clamped and free copies disagree, implementing a contrastive learning rule.

resistor networks) are relatively simple physical systems, they are able to solve classification[25] and allostery[26] tasks; when they are augmented with nonlinearity, they are in fact computationally universal[27]. The learning process is based on the contrastive learning rule[28]. In contrastive learning, two copies of the system are constructed. Both copies are excited with the same input. During training, the output of one of the copies is clamped to the target value, while the other copy is left free. Then, the parameters (springs) are updated proportionally to the difference between vibration amplitudes in the clamped and free copies (Fig. 1b). This learning process can be intuitively understood as changing the system parameters until the free copy—which has not been exposed to the correct answer—learns to mimic the copy that is clamped to the correct result.

The requirement for two copies of the system has been an obstacle to the realization of contrastive learning schemes[29]. Here, we overcome this challenge by relying on the crystalline symmetries of the metamaterial (Fig. 1c). When a lattice exhibits a symmetry, such as the four-fold rotational symmetry C_4 present in our case, eigenmodes corresponding to two-dimensional irreducible representations of the symmetry automatically appear in pairs[30]. Around the frequency of these eigenmodes, the system effectively behaves as two separate, identical oscillator networks. These two networks can be addressed independently by controlling the location at which excitation and clamping are applied. We will use these two symmetry-protected, identical copies to construct the free and clamped copies required by the contrastive learning rule. We will denote the modal displacements at site i by ψ_{i_x} and ψ_{i_y} respectively, and refer to both of them together as the computational field $\vec{\psi}$.

In self-learning mass-spring networks, the values of the springs connecting the computational degrees of freedom are also dynamical variables. To allow for the spring constants to vary, we encode them in an additional site eigenmode, that we refer to as the weight field σ_i. While the computational field $\vec{\psi}$ is different for every input sample—for example, for every flower being classified—the weight field σ_i accumulates all prior learned experience of the metamaterial. Thus, it requires a much longer memory time scale. We realize this long-term memory by parametrically driving the mode, modulating its local stiffness at two times the modal resonance frequency. When a system is

parametrically driven strongly enough, the system undergoes a bifurcation from which two stable, self-oscillating solutions emerge. These two stable solutions—characterized by opposite phases of oscillation—can be extremely long-lived, with transitions driven by thermal noise. Since the two stable states can be modeled by a spin-like binary variable, the weight field σ can be thought of as emulating an Ising model. Such emulators are commonly referred to as coherent Ising machines.

The self-learning metamaterial can be seen as a coherent Ising machine σ_i interacting with the computational field $\vec{\psi}_i$. Thus, we refer to it as a multifield coherent Ising machine. Throughout the rest of this section we will discuss how the learning rule emerges from nonlinear interactions between computational and weight fields. The nonlinearity plays a double role. First, it makes the springs that connect computational degrees of freedom dependent on the weight field σ. Second, it modulates the long-term memory of the weight field σ depending on the difference between clamped and free computational fields—causing the system to 'forget' the weights on sites where clamped and free copies disagree. We will finally show how these two interactions together allow the system to learn a classification problem.

The equations of motion of the material, describing the evolution of the fields $\vec{\psi}$ and σ at each site i are:

$$\ddot{\vec{\psi}}_i + \frac{\omega_c}{Q_c}\dot{\vec{\psi}}_i + w_c^2\vec{\psi}_i + \nabla_{\vec{\psi}_i} H_l = \vec{\xi}_{\psi,i}; \tag{1}$$

$$\ddot{\sigma}_i + \frac{\omega_l}{Q_l}\dot{\sigma}_i + \omega_l^2(1 + \alpha\sin(2\omega_l t))\sigma_i + \epsilon\sigma_i^3 + \nabla_{\sigma_i} H_l = \xi_{\sigma,i}. \tag{2}$$

We set the frequency of the computational and learning degrees of freedom to $\omega_c^2 = 0.5$ and $\omega_l^2 = 1$ respectively (the subindex is used to distinguish between computational and learning modes). The corresponding quality factors are set to $Q_c = 4000$ and $Q_l = 40$, while the parametric driving strength—how much the resonator frequency is externally varied over time—is set to $\alpha_l = 0.1$. The nonlinear Duffing coefficient plays a critical role in preventing the amplitude of the mode σ_i to grow without bounds, and is set to $\epsilon = 0.005$. The terms ξ represent the thermal noise exerted by the environment on the system. They take the usual form of a set of independent, Gaussian distributed random forces with autocorrelation $\langle\xi_{l/c}(t_0)\xi_{l/c}(t_1)\rangle = 2k_B T b_{c/l}\delta(t_1 - t_0)$. Interactions between degrees of freedom are captured by the learning potential $H_l = H_I + H_L$, where the local part H_L couples the fields locally at every site of the lattice—updating the long-term memory according to the learning rule—and the interaction part H_I couples each site with its nearest neighbors—ensuring that the field $\vec{\psi}$ processes information according to the weights stored in the σ field.

Single-site learning dynamics

In the proposed metamaterial, the learning rule arises due to local interactions at each site i. These interactions update the weight field (σ) when the free (ψ_x) and clamped (ψ_y) sub-systems disagree. In traditional contrastive learning, the weights are continuous variables, adjusted in proportion to the difference between clamped and free configurations $\psi_x - \psi_y$. However, in our model, every site of the weight field has only two stable phases of oscillation and, therefore, cannot be varied continuously. We adapt to this difference by implementing a probabilistic update process, where the probability that a weight σ_i flips increases with increasing $|\psi_x - \psi_y|$. To realize this probabilistic bit flipping, we rely on the fact that the memory of a parametric oscillator—the average time it takes for the state to flip randomly due to thermal noise—is highly dependent on the detuning[31]—the difference between its parametric excitation frequency and the optimal parametric resonance condition. We can thus realize probabilistic contrastive learning by applying a potential that shifts the natural frequency of σ_i as a function of $\psi_y - \psi_x$, causing the system to be detuned.

Such selective frequency shift can be accomplished with a cross-Kerr interaction. Cross-Kerr interactions shift the frequency of one mode as a function of the energy in another mode, and take the form

$$H_{L,i} = \mu(t)\psi_{x-y,i}^2\sigma_i^2 = \mu(t)(\psi_{x,i} - \psi_{y,i})^2\sigma_i^2. \tag{3}$$

Here, to turn a cross-Kerr interaction into a difference-sensitive detuning, we used the rotational symmetry of the site. Since the sites are symmetric, any combination between the eigenmodes ψ_x and ψ_y is also an eigenmode. Thus, we can look at the rotated set of modes $(\psi_{x+y}, \psi_{x-y}) = (\psi_x + \psi_y, \psi_x - \psi_y)$. In this new basis, the difference between modes $\psi_x - \psi_y$ can be seen as its own mode ψ_{x-y}. In an experimental setting, such a rotated cross-Kerr interaction could be realized by introducing a non-linear spring in a region of the resonator where the two modes ψ_x and ψ_y have opposite signs.

The potential in Eq. (3) breaks the C_4 rotational symmetry of the resonator; when this term is active, the system can no longer be seen as two separate copies. We overcome this by setting the strength $\mu(t)$ to zero during computation, preserving the two-copy picture. To initiate a weight update, we briefly turn on the cross-Kerr term, by increasing $\mu(t)$ according to a Gaussian pulse with peak value μ_m and a width Δ_t,

$$\mu(t) = \mu_m \exp\left\{\left(-\frac{(t - t_0)^2}{2\Delta_t^2}\right)\right\}. \tag{4}$$

Although here we chose to apply the learning potential $\mu(t)$ according to a Gaussian pulse, its specific shape is not critical. In Supplementary Note 2, we show that following a square pulse results in similar dynamics. Because $\mu(t)$ is independent of the state of the system, it is the same for every resonator and does not require a controller; its role is that of a system clock, indicating when learning shall occur.

Figure 2a shows the evolution of the energies in the modes σ, ψ_x and ψ_y during a weight update. The resonator is initialized with specific values of the fields $\vec{\psi}$ and σ. During operation, the initial condition for $\vec{\psi}$ is set by the interaction with the neighbors, as will be discussed later in the paper. Our simulations reveal that, when the learning potential is applied, the oscillation of the weight degree of freedom (σ) decreases with increasing difference between initial conditions of ψ_x, ψ_y. This is a consequence of the increased detuning of σ, which drives the mode away from resonance. The lower amplitude of the detuned mode makes thermally induced flips more likely, thus reducing the memory of the system. When the learning potential is active, we also observe a periodic exchange of energy between free and clamped modes. This is an expected consequence of H_L breaking the fourfold rotational symmetry—introducing a difference between the $x + y$ and $x - y$ directions and thus distorting the picture where the clamped and free copies are independent.

The difference between clamped and free oscillations affects the probability of flipping a weight in a step-like fashion (Fig. 2b). This step-like response arises because σ_i only self-oscillates in a specific region of parametric excitation frequencies and amplitudes (Fig. 2c). When the difference between clamped and free oscillators is large enough, the nonlinear interaction (Eq. (3)) shifts this region outside of the excitation conditions. As a consequence, the system ceases to self-oscillate and forgets its state. At low temperature, this phenomenon is abrupt—as long as the excitation remains within the self-oscillation region, the system remembers its state. In contrast, when the temperature is finite, random thermal flips start to occur at finite amplitudes, smoothing out the response (Fig. 2b). The difference $\psi_x - \psi_y$ at which the forgetting transition occurs is a tunable parameter. We can control it by adjusting the strength of the interaction μ_m (Fig. 2d). This state-dependent switching relies on the ratio between the quality factors of the computational and learning degrees of freedom. Because outside parametric resonance the computational degree of freedom has a much longer memory lifetime than the weight degree of freedom, the difference between clamped and free copies remains throughout the update protocol. In a practical scenario, this difference could be implemented by controlling the coupling between each mode and the environment, or via external cooling signals to speed up the learning dynamics.

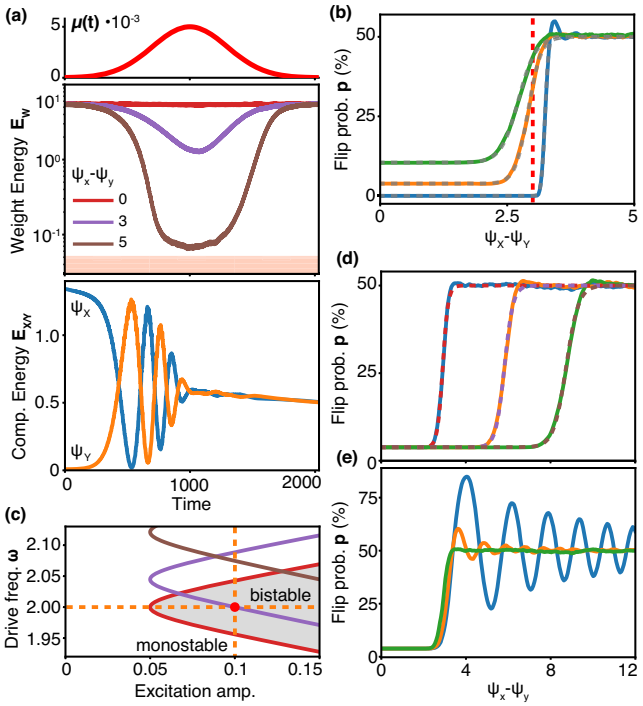

Fig. 2 | Single-site learning dynamics. a Evolution of the energies in the weight (σ, middle panel) and computational (ψ_x, ψ_y, bottom panel) resonators, as a learning pulse is applied (top panel). **b** Probability of flip during a learning protocol as a function of $\psi_X - \psi_Y$ for different temperatures, corresponding to k_BT values of 0.001 (blue), 0.05 (orange) and 0.1 (green). **c** Parametric self-oscillation region (shaded gray area) of the weight degree of freedom σ (See Supplementary Note 1 for the analytical derivation of the parametric resonance region); in the shaded region, the weight is bistable and thus has long term memory. The excitation conditions are shown as dashed orange lines and red dot. When $\psi_X - \psi_Y$ is not zero, applying the learning potential shifts the self-oscillation region. The value of $\psi_x - \psi_y$ where σ goes out of parametric resonance is shown as a red dashed line in (**b**). **d** Probability of spin flip as a function of $\psi_x - \psi_y$, for learning potentials $\mu_m = \mu_{m,0}/n^2$ with $n = 1$ (blue), $n = 2$ (orange) and $n = 3$ (green), illustrating how the step location can be shifted by setting the potential. **e** Probability of spin flip as a function of the protocol durations $\Delta_t = 50$ (blue), $\Delta_t = 100$ (orange), and $\Delta_t = 300$ (green), showing the emergence of probability oscillations at short learning pulses. The dashed curves in (**c**, **d**) show the fit with Eq. (5). Throughout the figure, the parameters are $2\omega_l = 2$, $\omega_l^2\alpha = 0.1$, $\mu_{m,0} = 0.005$, and the step duration is $\Delta_T = 300$ unless stated otherwise.

We empirically observe that the probability of transitions p closely follows (Fig. 2b and d) a hyperbolic tangent relation,

$$p = p_0 + \frac{1-2p_0}{4}[1 + \tanh(\beta(|\psi_{x-y}| - \psi_T))], \tag{5}$$

in which ψ_T is the threshold for bit flipping, β determines the slope of the flipping transition, and p_0 is the probability of spontaneous thermal flipping in absence of contrast between clamped and free configurations. This is unsurprising, because our weight field behaves as an effective spin system, and hyperbolic tangents commonly arise in spin-flip probabilities due to detailed balance considerations, as exemplified paradigmatically by Glauber dynamics[32]. Later, Eq. (5) will allow us to predict the probability of bit flips in large lattices without having to simulate expensive stochastic differential equations for each site. Remarkably, the sigmoidal description breaks down for shorter pulse widths Δ_t. In these cases, the oscillation of σ does not have time to decay below the thermal noise, thus forgetting its state. The remaining amplitude is phase-shifted by the learning potential, giving rise to oscillations of the bit-flip probability as a function of $\psi_x - \psi_y$ (Fig. 2e) when the accumulated shift is a multiple of π. In the rest of this work, we will operate in the sigmoidal regime—corresponding to long learning pulses—to achieve a monotonic relation between weight flip probability and clamped-free difference.

Site-site interactions

In machine learning models, the role of the weights is to determine what computation is carried out, by setting the strength of the coupling between neurons. Consequently, in a self-learning physical system, storing weight and computational degrees of freedom separately is not sufficient; we must also ensure that the weights influence the computation that is being performed. In our metamaterial, this influence is established through a nonlinear interaction H_I acting between each pair of degrees of nearest-neighbor sites i, j. It is governed by an energy of the form

$$H_I = -\lambda(t)[c_0 + (\sigma_i - \sigma_j)^2]\vec{\psi}_i^{\,T}\vec{\psi}_j, \tag{6}$$

This term generates the effective coupling springs that connect neighboring computational degrees of freedom ($\psi_{x,i} \leftrightarrow \psi_{x,j}$ and $\psi_{y,i} \leftrightarrow \psi_{y,j}$). Depending on the phase of the weight degrees of freedom, these coupling springs can take two values (Fig. 1)—determined by $\lambda(t) = \lambda_m$ and c_0; if the weight field has the same phase of oscillation in both sites, the spring constant will be $k_{\text{low}} = \lambda_m c_0$, while if weights oscillate at opposing phases, the spring constant will be $k_{\text{high}} = \lambda_m(c_0 + 4 < \sigma^2 >)$. This effective spring model is valid as long as the computational amplitude remains small and the potential in Eq. (6) does not alter the weight amplitude. Similarly to $\mu(t)$, $\lambda(t)$ is a state-independent signal that is identical for all resonators, acting as a system clock. In Supplementary Note 3, we discuss how H_I could be implemented in terms of cubic springs.

Although each individual weight σ_i is binary, in metamaterials composed of multiple sites (Fig. 3a) the transmitted energy between input and output can take many different values, depending on the global weight configuration. This can be seen in the density of states, which quantifies the number of weight configurations that result in a lattice transmissivity around a specific value (Fig. 3b and c). This distribution depends on lattice size and on the $k_{\text{high}}/k_{\text{low}}$ ratio, which is chosen (approximately, by trial and error) to have the transmissivities spread out as uniformly as possible. Because the lattice can be trained to approximate continuous transmissivities, as long as they fall within the range of expressible values, we treat this element as a single programmable weight—which we refer to as a learnistor. Networks of trainable weights, combined with fixed nonlinear activation functions, are computationally universal[33]; thus, we hypothesize that such trainable elements can form building blocks for more advanced learning architectures.

To perform inference with the metamaterial, we encode the input in the amplitude of a harmonic force, applied to both computational degrees of freedom, $\psi_{x,\text{input}}$ and $\psi_{y,\text{input}}$, at the input site. We let the lattice reach a steady-state, and read the result in the amplitude of the output computational degree of freedom, $\psi_{x,\text{output}}$. During this process, we keep $\lambda(t)$ at its inference value λ_m and $\mu(t)$ at zero. To perform a training iteration, in addition to the input forces, we also clamp the output site $\psi_{y,\text{output}}$ to the desired value. Then, once the lattice reaches a steady-state, we perform a weight update by increasing $\mu(t)$ according to the protocol discussed in the previous section. During the weight update, we momentarily set $\lambda(t) = 0$, turning the lattice into a set of disconnected sites. Under these conditions, the learning protocol $\mu(t)$ introduces a probabilistic weight flip, whose initial condition is given by the steady-state amplitude of the computational degrees of freedom. Figure 3d and e illustrates the training of a metamaterial to match a specific signal transmissivity. At every training iteration, the expected value of the output energy gets closer to the target value, while the variance between different runs decreases.

Learning the Iris dataset

We illustrate the capabilities of the proposed system on the Iris classification test[34]. The dataset contains measurements of the petal length l_p, petal width w_p, sepal length l_s and sepal width w_s (Fig. 4a) for a set of 150 flowers belonging to the species *iris setosa*, *iris virginica*, and *iris versicolor*. The

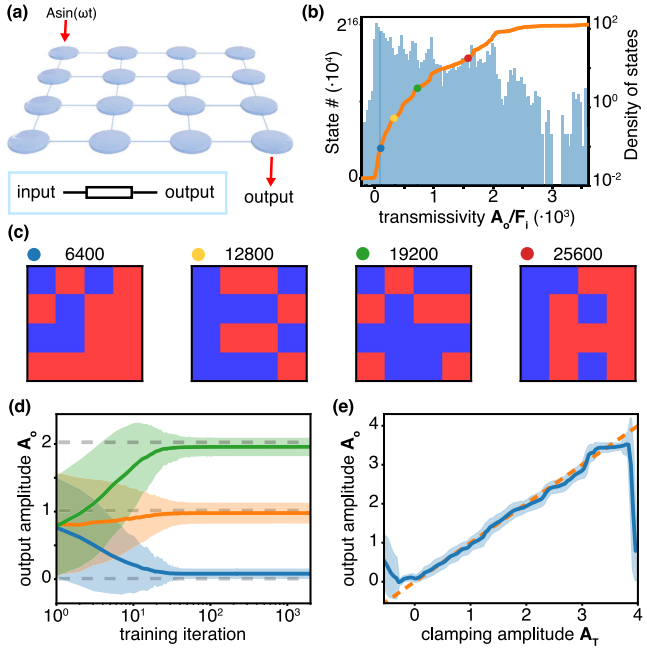

Fig. 3 | Site-site interactions in a lattice. a Example metamaterial, consisting of a 4 × 4 lattice. A harmonic force with frequency ω_c and amplitude F_i is applied to both clamped and free degrees of freedom at the input site, situated at the top-left corner. The output amplitude A_o is measured at the bottom-right corner. **b** Cumulative density of states (orange) and density of states (blue) as a function of the transmissivity, computed via the effective mass-spring model. The dots correspond to simulations of the transmissivity using the full nonlinear system (Eq. (2)) with an excitation force of 10^{-4}. **c** Weight configurations corresponding to the dots in (**b**). **d** Evolution of the transmissivity as a function of training iteration for target values A_T of 0, 1 and 2 (dashed line). **e** Output amplitude A_o as a function of the clamping amplitude, after 200 learning iterations, starting from a random configuration. The shaded area represents one standard deviation. The dashed orange line corresponds to an ideal learning response. We observe that, for transmissivity values for which a weight configuration exists, the transmitted amplitude approximately converges to the clamping amplitude. **d**, **e** have been computed by averaging 2000 training runs, with the shaded area representing the standard deviation.

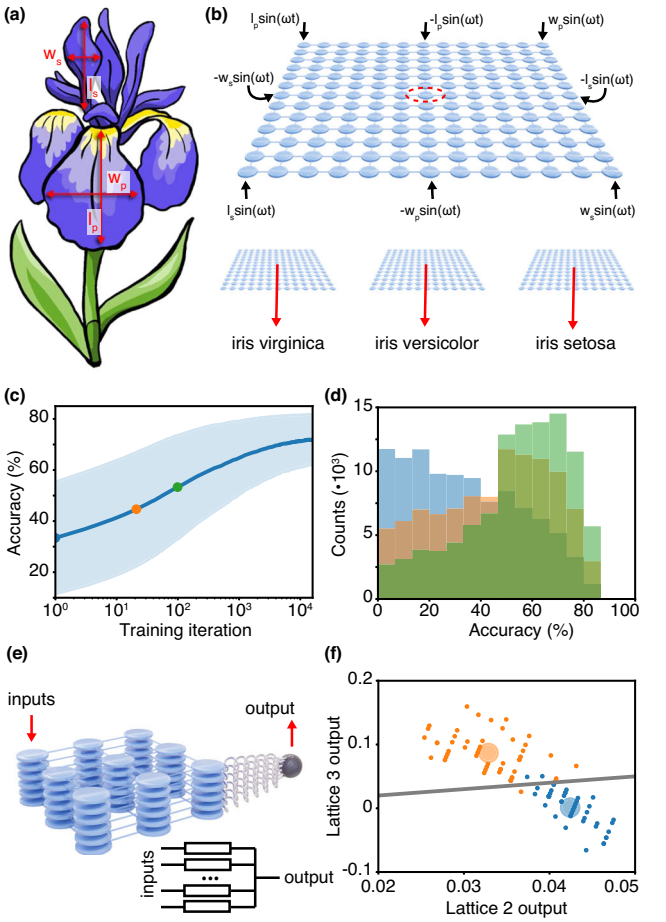

Fig. 4 | Iris flower classification using a multifield coherent Ising machine. **a** Flower features contained in the Iris dataset. **b** The features are injected into the multifield coherent Ising machine, by encoding them in the amplitude of harmonic excitations at the computational frequency ω_c. The output is taken at the central site. Positive and negative copies of the signals are applied, as the lattice cannot perform subtractions. The full model consists of three multifield Coherent Ising Machines, corresponding to each of the model classes. The machines learn to produce a high amplitude when excited by a sample of their corresponding class. **c** Evolution of the mean classification accuracy during training. The shaded area corresponds to one standard deviation. **d** Histograms of the classification accuracy computed on an untrained lattice (blue), and after 20 (orange) and 98 (green) training iterations. The training times corresponding to the histograms are indicated as solid dots in (**c**). **e** Architecture consisting of six lattices connected to an output site. Lattices are driven with l_w, l_p, $-l_w$, $-l_p$, $+b$ and $-b$ respectively, where $b = 3$ is a model bias. **f** Predicted outputs for the optimal spin texture when excited by iris versicolor (blue) and setosa (orange). The circles represent the mean training objective.

classifier consists of a system of three 13 × 13 lattices; the lattice with the highest output amplitude, measured at the central site, will be the inferred flower species. We encode the geometric features (l_p, w_p, l_s, w_s) in the amplitudes of a set of harmonic excitation forces acting on each lattice. These signals are applied at sites through the boundary of the lattice (Fig. 4b), acting on both free ψ_x and clamped ψ_y computational modes. The input and output locations are selected so that the output is at an approximately equal distance from all inputs. Since the lattices cannot produce large negative transmissivities (Fig. 3b), we apply both positive and negative copies of each feature signal. Alternative interaction potentials can actually produce negative transmissivities, as is discussed in the Supplementary Note 4.

The training process starts by initializing each of the three lattices with a random weight configuration. Then, for every learning iteration, we subject all lattices to the forces encoding the features of a randomly chosen flower. We clamp the output site $\psi_{y,\text{center}}$ to 55 for the lattice corresponding to the current type of flower, and to − 35 for the lattices corresponding to the other flowers (these values have been chosen empirically to fall within the range of expressible values). Finally, we apply a weight update protocol to all lattices, consisting of setting the coupling λ to zero, and changing $\mu(t)$ following a Gaussian pulse. This process is repeated for 2000 iterations.

We observe that the system starts from an accuracy of 33.2 ± 22.1%, as one would expect from random choice. As more training iterations are performed, the accuracy increases to 71.9 ± 9.8%, providing a clear signature of physical learning. These results are remarkable as the system stores only 156 bits of information, contains only Kerr nonlinearities, and is directly trained by exposing it to the desired output. Using a more advanced training algorithm, where the learning process is stopped when the precision exceeds a threshold, we achieve 80.0 ± 7.0% accuracy, with the best-performing runs reaching 87.3%. Although these accuracy figures are much lower than traditional digital machine learning algorithms (methods such as k-nearest neighbors, support vector machines and decision trees achieve performances in the 93.0 − 97% range), and to electronic implementations of learning rules (which can already reach values around 95%), they are remarkable in that they are achieved through the dynamical evolution of a nonlinear mass-spring network (subject to examples and simple clocking pulses). Our accuracies are however in line with similar 'unconventional learning' demonstrations, such as classifying with origami sheets (which reach best-values of 92% on a simpler two-class flower test) or when using

EEG signals (which achieve accuracies 80%). To confirm that our classification provides evidence of learning, and is not a result of random fluctuations, we perform statistical hypothesis testing in Supplementary Note 5. Our numerical simulations also reveal that the system is tolerant to realistic amounts of experimental disorder (such prospective experimental considerations are explored in Supplementary Notes 6 and 7).

Although in line with other unconventional learning results, the limited accuracy of our trained classifier merits further investigation. We identify two possible limiting factors: The binary nature of the weights in our system—which may reduce the model expressivity, or peculiarities of the training protocol. To separate the two effects, we construct a digitally-trained multiclass linear architecture using networks of 3 × 3 resonator lattices, which are highly quantized. The model projects a feature vector $\vec{f} = (p_l, p_w, 1)$ onto a set of weight vectors corresponding to the three flower classes $\vec{w}_{\text{virginica}}, \vec{w}_{\text{versicolor}}, \vec{w}_{\text{setosa}}$. Every class is implemented using a network of three pairs of 3 × 3 lattices connected to an output mass (Fig. 4e), with each pair of lattices expressing the positive and negative values of a specific weight vector element. Using such architecture, as opposed to a large lattice, allows us to identify specific sub-lattices with model coefficients. We initialize the spin texture of every lattice to the closest value required by the linear model. When simulating the lattice dynamics, the model achieves an accuracy of 97.3%—the maximum possible for a linear classifier. This indicates that weight quantization cannot explain the observed performance degradation

Therefore, the low precision on the Iris dataset must originate in the training dynamics. We know from Fig. 3b, that lattices can be trained to arbitrary transmissivities within their range of expressivity. This works because spin changes stop taking place when the transmitted amplitude matches the target. However, when training on the Iris dataset, each sample causes the lattice to shift towards a different transmissivity (Fig. 4f), preventing the lattice from settling into a stable value. In prior, high-accuracy, electronic realizations of contrastive learning[23], the training process has not prescribed the output to a specific value, but instead *nudged* incorrect answers towards the correct value. We hypothesize that such sophisticated training methods may enable the system to achieve higher accuracies—potentially comparable to electronic linear classifiers, by allowing it to settle into high-accuracy configurations even if the output does not match a specific transmitted amplitude. However, evaluating such more complex protocols falls outside the scope of this work, which focuses on control-free systems, i.e., where the input signals applied to the network are independent of its state.

Discussion

We have shown that a metamaterial composed of a network of parametrically-driven, multimodal resonators can learn to solve a classification task, when excited with example signals. The material performs its own weight update according to a learning rule, requiring only two simple drive clock signals $\lambda(t)$ and $\mu(t)$ in addition to the data and labels. However, the metamaterial does not require independent controllers adjusting each individual weight. The metamaterial learning dynamics is based on contrastive learning, which involves comparing two copies of the system—one left free, and one clamped to the desired output—and updating the model weights according to the difference between copies. However, unlike traditional contrastive learning, our model weights are binary and cannot be updated continuously. We circumvent this limitation by reformulating the learning rule as a probabilistic process, where the probability of a weight flip is a function of the mismatch between the two copies. This mismatch-dependent flip rate is reminiscent of evolutionary mechanisms in bacteria, where the mutation rate increases when the organism is under stress[35–37]. Similarly, our system increases the weight mutation rate at sites where its clamped and free copies disagree.

In our model, learning emerges from the interplay of symmetry, multistability, and noise, features not typically associated with computation. This work therefore, contributes to the growing evidence that such physical effects hold significant, untapped potential for information processing[38–41]. In fact, symmetry plays an essential role in the learning process: Clamping the ψ_y breaks the symmetry between the two components of the computational field $\vec{\psi}$. The weight field, σ, then evolves to restore this broken symmetry over time. As symmetry is gradually restored, the free part of the system learns to mimic the clamped part, not by accessing the training data directly, but through persistent changes in the weight variable σ. This perspective suggests a novel symmetry-based design principle for self-learning materials, much like symmetry considerations guide the design of emergent condensed matter phenomena such as topological edge states[42]. The presence of multiple time scales is also an essential component. These are important in the brain[43] and can be replicated in artificial systems through parametric drive.

By providing a tractable physical model for emergent learning—analogous to the tight-binding Hamiltonians that describe topological insulators and superconductors—our work invites condensed matter-inspired questions about learning. For instance, one can now investigate the influence of disorder[44,45], aging[46], and phase transitions[47–49] on learning behavior. Although the present work is concerned with non-dimensional tight-binding models and thus no specific performance claims can be made, our model may inspire new directions to train machine learning models with physical computers, as the constituent oscillators can be realized with components that operate at optical frequencies[50], consume little energy[51], or are integrated with very high densities[52]. Our work also poses novel exciting fundamental questions, for example involving learning dynamics in discrete spin systems, architectural and scaling considerations, as well as addressing the experimental realization of such models.

Methods

Simulation of the single-site dynamics

We solve the stochastic differential equations (SDEs) using a Splitting Path Runge-Kutta solver (SPaRK), while for the deterministic simulations (four dots in Fig. 3b), we use a Dormand-Prince algorithm with order 7-8. Both solvers are implemented in Python using the Diffrax library. We choose a time step for a minimum resolution of 50 points per period of oscillation.

Simulation of the learning process

We simulate the learning process by computing the equilibrium amplitudes of $\vec{\psi}$ using an effective, linearized mass-spring model for the computational degrees of freedom. The effective model is calculated around the resonance frequency ω_c. The coupling springs are given by the linearized version of Eq. (6), while the diagonal terms are governed by the damping—in resonance, local inertia cancels with local stiffness—and take the form $i\omega_c^2/Q_c$. Once the computational field has been calculated using the linearized model, we flip the weights according to the single-site flip probability (Eq. (5)), based on the difference between clamped and free fields. The single-site flip probability provides an accurate result as we switch off the inter-site potential during coupling, thus reducing the model to a set of independent elements.

For the learning examples of Figs. 3 and 4, we use the parameters $k_{\text{high}} = 10\omega_c^2/Q_c$, $4k_{\text{low}} = k_{\text{high}}$, $\beta = 0.273$ and $\psi_T = 55$. In Fig. 3, we use the flip probability parameters $\beta = 750$ and $\psi_T = 0.16$, while in Fig. 4, the parameters are $\beta = 0.273$ and $\psi_T = 55$.

To impose the boundary conditions, we divide the degrees of freedom into internal (subscript I) and output (subscript o) components. We write the system $\mathbf{Du} = \mathbf{f}$ in block form as:

$$\begin{bmatrix} \mathbf{D}_{II} & \mathbf{d}_{Io} \\ \mathbf{d}_{oI}^T & d_{oo} \end{bmatrix} \begin{bmatrix} \mathbf{u}_I \\ u_o \end{bmatrix} = \begin{bmatrix} \mathbf{f}_I \\ f_o \end{bmatrix} \tag{7}$$

where u_o is the prescribed output degree of freedom, f_o is the constraint force, and F_I is the external force (which inputs the data into the system). From the first block row, we bring the prescribed output to the right-hand side to obtain the reduced system $\mathbf{D}_{II}\mathbf{u}_I = \mathbf{f}_I - \mathbf{d}_{Io}u_o$, which we solve for the unknown internal DOFs $\mathbf{u}_I$. Note that in the reduced system equation, it is not necessary to explicitly calculate the constraint force f_o to simulate the system dynamics.

In an experimental setting, the boundary conditions could be applied by connecting the output degree of freedom to a very large mass, or by applying external feedback.

Data availability

No datasets were generated or analysed during the current study.

Code availability

The codes used in this study are available from the corresponding authors upon reasonable request.

Received: 2 September 2025; Accepted: 9 December 2025;
Published online: 31 January 2026

References

1. Rajabalipanah, H., Momeni, A., Rahmanzadeh, M., Abdolali, A. & Fleury, R. Parallel wave-based analog computing using metagratings. *Nanophotonics* **11**, 1561 (2022).
2. Hu, J. et al. Diffractive optical computing in free space. *Nat. Commun.* **15**, 1525 (2024).
3. Wang, T. et al. An optical neural network using less than 1 photon per multiplication. *Nat. Commun.* **13**, 123 (2022).
4. Silva, A. et al. Performing mathematical operations with metamaterials. *Science* **343**, 160 (2014).
5. Cordaro, A. et al. High-index dielectric metasurfaces performing mathematical operations. *Nano Lett.* **19**, 8418 (2019).
6. Cordaro, A. et al. Solving integral equations in free space with inverse-designed ultrathin optical metagratings. *Nat. Nanotechnol.* **18**, 365 (2023).
7. Hughes, T. W., Williamson, I. A., Minkov, M. & Fan, S. Wave physics as an analog recurrent neural network. *Sci. Adv.* **5**, eaay6946 (2019).
8. Dubček, T. et al. In-sensor passive speech classification with phononic metamaterials. *Adv. Funct. Mater.* **34**, 2311877 (2024).
9. Ashtiani, F., Geers, A. J. & Aflatouni, F. An on-chip photonic deep neural network for image classification. *Nature* **606**, 501 (2022).
10. Lin, X. et al. All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004 (2018).
11. Oguz, I. et al. Optical diffusion models for image generation. *Adv. Neural Inf. Process. Syst.* **37**, 59150 (2024).
12. Bösch, C., Roeder, G., Serra-Garcia, M. & Adams, R.P. Local learning rules for out-of-equilibrium physical generative models. Preprint at https://doi.org/10.48550/arXiv.2506.19136 (2025).
13. Luce, A., Alaee, R., Knorr, F. & Marquardt, F. Merging automatic differentiation and the adjoint method for photonic inverse design. *Mach. Learn. Sci. Technol.* **5**, 025076 (2024).
14. Bordiga, G. et al. Automated discovery of reprogrammable nonlinear dynamic metamaterials. *Nat. Mater.* **23**, 1486 (2024).
15. Cui, T. J., Qi, M. Q., Wan, X., Zhao, J. & Cheng, Q. Coding metamaterials, digital metamaterials and programmable metamaterials. *Light Sci. Appl.* **3**, e218 (2014).
16. Hebb, D.O. *The organization of behavior: A neuropsychological theory* (John Wiley and Sons, 1949).
17. Milner, P. A brief history of the Hebbian learning rule. *Can. Psychol. Psychologie Canadienne* **44**, 5 (2003).
18. Wright, W. J., Hedrick, N. G. & Komiyama, T. Distinct synaptic plasticity rules operate across dendritic compartments in vivo during learning. *Science* **388**, 322 (2025).
19. Scellier, B. & Bengio, Y. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Front. Comput. Neurosci.* **11**, 24 (2017).
20. Momeni, A., Rahmani, B., Malléjac, M., Del Hougne, P. & Fleury, R. Backpropagation-free training of deep physical neural networks. *Science* **382**, 1297 (2023).
21. López-Pastor, V. & Marquardt, F. Self-learning machines based on Hamiltonian echo backpropagation. *Phys. Rev. X* **13**, 031020 (2023).
22. Altman, L. E., Stern, M., Liu, A. J. & Durian, D. J. Experimental demonstration of coupled learning in elastic networks. *Phys. Rev. Appl.* **22**, 024053 (2024).
23. Dillavou, S., Stern, M., Liu, A.J. & Durian, D.J. Demonstration of decentralized physics-driven learning. *Phys. Rev. Appl.* **18**, 014040 (2022).
24. Stern, M., Guzman, M., Martins, F., Liu, A. J. & Balasubramanian, V. Physical networks become what they learn. *Phys. Rev. Lett.* **134**, 147402 (2025).
25. Li, S. & Mao, X. Training all-mechanical neural networks for task learning through in situ backpropagation. *Nat. Commun.* **15**, 10528 (2024).
26. Rocks, J. W. et al. Designing allostery-inspired response in mechanical networks. *Proc. Natl. Acad. Sci. USA* **114**, 2520 (2017).
27. Scellier, B. & Mishra, S. Universal approximation theorem for nonlinear resistive networks. *Phys. Rev. Appl.* **23**, 044009 (2025).
28. Stern, M., Hexner, D., Rocks, J. W. & Liu, A. J. Supervised learning in physical networks: From machine learning to learning machines. *Phys. Rev. X* **11**, 021045 (2021).
29. Anisetti, V. R., Kandala, A., Scellier, B. & Schwarz, J. Frequency propagation: multimechanism learning in nonlinear physical networks. *Neural Comput.* **36**, 596 (2024).
30. Dresselhaus, M.S., Dresselhaus, G. & Jorio, A. *Group theory: application to the physics of condensed matter* (Springer Science & Business Media, 2007).
31. Dykman, M., Maloney, C., Smelyanskiy, V. & Silverstein, M. Fluctuational phase-flip transitions in parametrically driven oscillators. *Phys. Rev. E* **57**, 5202 (1998).
32. Glauber, R. J. Time-dependent statistics of the Ising model. *J. Math. Phys.* **4**, 294 (1963).
33. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* **2**, 303 (1989).
34. Fisher, R. A. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **7**, 179 (1936).
35. Bjedov, I. et al. Stress-induced mutagenesis in bacteria. *Science* **300**, 1404 (2003).
36. Gutierrez, A. et al. β-lactam antibiotics promote bacterial mutagenesis via an rpos-mediated reduction in replication fidelity. *Nat. Commun.* **4**, 1610 (2013).
37. Pribis, J. P., Zhai, Y., Hastings, P. & Rosenberg, S. M. Stress-induced mutagenesis, gambler cells, and stealth targeting antibiotic-induced evolution. *MBio* **13**, e01074 (2022).
38. Aifer, M. et al. Thermodynamic linear algebra. *npj Unconv. Comput.* **1**, 13 (2024).
39. Melanson, D. et al. Thermodynamic computing system for ai applications. *Nat. Commun.* **16**, 3757 (2025).
40. Harabi, K.-E. et al. A memristor-based Bayesian machine. *Nat. Electron.* **6**, 52 (2023).
41. Chen, D., Kent, A. D., Sels, D. & Morone, F. Solving combinatorial optimization problems through stochastic Landau-Lifshitz-Gilbert dynamical systems. *Phys. Rev. Res.* **7**, 013129 (2025).
42. Altland, A. & Zirnbauer, M. R. Nonstandard symmetry classes in mesoscopic normal-superconducting hybrid structures. *Phys. Rev. B* **55**, 1142 (1997).
43. Masset, P. et al. Multi-timescale reinforcement learning in the brain. *Nature* **642**, 682 (2025).
44. Zu, M., Desai, A. & Goodrich, C. P. Fully independent response in disordered solids. *Phys. Rev. Lett.* **134**, 238201 (2025).
45. Krushynska, A. O. & van Hecke, M. Multi-frequency acoustic steering in rationally pruned disordered networks. *Matter* **8**, 102364 (2025).
46. Dohare, S. et al. Loss of plasticity in deep continual learning. *Nature* **632**, 768 (2024).
47. Kirkpatrick, S. & Selman, B. Critical behavior in the satisfiability of random Boolean expressions. *Science* **264**, 1297 (1994).
48. Amit, D. J., Gutfreund, H. & Sompolinsky, H. Spin-glass models of neural networks. *Phys. Rev. A* **32**, 1007 (1985).

49. Power, A., Burda, Y., Edwards, H., Babuschkin, I. & Misra, V. Grokking: generalization beyond overfitting on small algorithmic datasets. Preprint at https://doi.org/10.48550/arXiv.2201.02177 (2022).
50. Okawachi, Y. et al. Demonstration of chip-based coupled degenerate optical parametric oscillators for realizing a nanophotonic spin-glass. *Nat. Commun.* **11**, 4119 (2020).
51. Cupertino, A. et al. Centimeter-scale nanomechanical resonators with low dissipation. *Nat. Commun.* **15**, 4255 (2024).
52. Zahedinejad, M. et al. Two-dimensional mutually synchronized spin hall nano-oscillator arrays for neuromorphic computing. *Nat. Nanotechnol.* **15**, 47 (2020).

Acknowledgements

We are thankful to Menachem Stern, Hermen-Jan Hupkes, Cyrill B˘osch, Henrik Wolf, Tena Dub˘cek and Martin van Hecke for helpful discussions. The illustrations in Figs. 1, 3a, and 4a and b have been provided by Laura Canil from Canil Visuals.

Author contributions

D.d.B. performed numerical simulations and analyzed data. M.S-G. conceptualized the research and contributed to data analysis. Both authors contributed to the writing and reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s44455-025-00015-4.

Correspondence and requests for materials should be addressed to Marc Serra-Garcia.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025